

REMARKS

I. Introduction

In response to the Office Action dated July 13, 2005, claims 3, 22, and 41 have been cancelled, and claims 1, 4, 6, 9, 10, 12, 13, 14, 20, 23, 25, 28, 31, 32, 33, 39, 42, 44, 47, 48, 50, 51, and 52 have been amended. Claims 1-2, 4-21, 23-40, and 42-57 remain in the application. Re-examination and re-consideration of the application, as amended, is requested.

II. Non-Art Rejections

In paragraph (3) of the Office Action, the specification was objected to for an informality relating to the issued patent numbers cross-referenced in the patent application.

Applicants have amended the specification to overcome this objection.

III. Prior Art Rejections

In paragraphs (4)-(5) of the Office Action, claims 1-4, 8-13, 15-16, 18-23, 34-35, 37-42, 47-51, 53-54, and 56-57 were rejected under 35 U.S.C. §103(a) as being unpatentable over Knight et al., U.S. Patent No. 6,571,234 (Knight). In paragraph (6) of the Office Action, claims 5, 24, and 43 were rejected under 35 U.S.C. §103(a) as being unpatentable over Knight in view of O'Brien, U.S. Patent No. 6,055,569 (O'Brien). In paragraph (7) of the Office Action, claims 6-7, 14, 25-26, 33, 44-46, and 52 were rejected under 35 U.S.C. §103(a) as being unpatentable over Knight in view of Borman et al., U.S. Patent No. 6,226,655 (Borman). In paragraph (8) of the Office Action, claims 17, 36, and 55 were rejected under 35 U.S.C. §103(a) as being unpatentable over Knight in view of Batchelder et al., U.S. Patent No. 6,351,767 (Batchelder).

Specifically, the independent claims were rejected as follows:

1, 9, 20, 28, 39, and 47

In regard to independent Claim 1 (and similarly independent Claims 9, 20, 28, 39, and 47), Knight teaches (a) *requesting a first web page* (Col. 12, lines 61-67; Fig. 3B) wherein a user accesses a main web site with a browser, and is presented with a web page (e.g., Fig. 3B).

Knight continues by teaching (b) *retrieving the first web page in a browser in response to the request, wherein the first web page comprises an applet tag in that the message board interface 300 is implemented as part of a program based on a Java applet that is downloaded into the users computing system by server 220* (Col. 11, lines 21-25; Col. 12, lines 61-67). It is notoriously well known in the art of web design that to affect the downloading of an applet from a server to a client, one typically places a reference to the applet (in an applet tag) into the web page to which it is associated with. The browser processes the <APPLET> tag contents and downloads and executes the applet. Hence, Knight also

teaches (c) requesting an applet identified by the applet tag; (d) receiving the applet; (e) executing the applet, wherein the applet is configured to).

Knight further teaches (1) request one or more web objects that are likely to be accessed next as part of one or more additional web pages that are likely to be requested by a user (Col. 23, lines 26-67; Col. 24, lines 1-36). The messages, in this case, qualify as the claimed "web objects", since they are accessed, downloaded, etc. from a web site run by a web server to a web client (browser) executing a Java applet, which performs the actual interactions.

Knight also teaches (2) receive the one or more requested web objects; (3) pre-cache the one or more requested web objects by copying the one or more requested web objects into a cache of the browser in that messages are off-loaded from the message server to the message browser on the client. The user's data (and other popularly retrieved data) is pre-cached, pre-compressed, and downloaded to his/her local machine where it can be handled far faster, and without delays caused by other user transactions. Thus, the data is strategically located to areas where there are more computing resources to process it according to the user's requirements (Col. 24, lines 34-36).

Knight teaches (f) receiving a request from the user for a second web page in that a client applet interface is executed from within a browser that allows the user to interact with multiple messages (Col. 11, lines 21-25; Col. 12, lines 61-67). Making a request for the next mail message would have been initiated through the interface. Having previously cached messages that were either pre-designated by the user, or anticipated by the software as being likely to be accessed, the request for the next message would have looked first to the cache, and then displayed it for the user (Col. 13, lines 5-45). Though Knight does not explicitly teach requesting a web page, it would have been obvious to one of ordinary skill in the art at the time of the invention to conclude that requesting and receiving messages through a web-based applet client, from the client side would have been functionally equivalent to that which is claimed. One benefit would have been to minimize the number of actual query transactions on the server, because the user's data (and other popularly retrieved data) is pre-cached, pre-compressed, and downloaded to his/her local machine where it can be handled far faster, and without delays caused by other user transactions. Thus, the data is strategically located to areas where there are more computing resources to process it according to the user's requirements.

Likewise, Knight would have performed the equivalent of (g) retrieving the web object from the cache for the second web page in response to the request from the user; and (h) displaying the retrieved web object, as part of the second web page, in the browser using the applet interface to the web message board, since these functions already exist in the interface (Col. 11, lines 21-25; Col. 12, lines 61-67).

Applicants have amended the claims to incorporate the limitations of prior claims 3, 22, and 41 which were rejected as follows:

In regard to dependent claim 3 (and similarly dependent Claim 22, 41), Knight teaches that the applet is further configured to receive an object list of the one or more web objects likely to be accessed next and wherein the request for the one or more web objects is for one or more web objects in the object list in that a query handling routine automatically downloads those messages corresponding to information categories previously indicated as of most interest to the particular user. This can occur at the beginning of a session, or during idle periods, so that the user is given a faster response time for messages of interest to him/her (Col. 6, lines 18-24).

Applicants traverse the above rejections for one or more of the following reasons:

- (1) Knight, O'Brien, Borman, and Batchelder do not teach, disclose or suggest an object list;

(2) Knight, O'Brien, Borman, and Batchelder do not teach, disclose or suggest an object list that identifies web objects likely to be accessed next as part of an additional web page likely to be requested by a user; and

(3) Knight, O'Brien, Borman, and Batchelder do not teach, disclose or suggest requesting objects identified in an object list.

Amended independent claims 1, 9, 20, 28, 39, and 47 are generally directed to pre-caching information in a browser cache. Specifically, a client initially requests a first web page from a server. In response, the server sends back the first web page containing an applet. The applet is executed by the client and initially obtains an object list. The object list identifies web objects that are likely to be accessed next as part of additional web pages that the user/client is likely to request. The applet is then configured to request one or more web objects that are identified in the object list. The web objects are sent by the server and cached in the browser cache on the client. The client then receives a request for a second web page from the user. In response to the request, the client retrieves the web objects from the cache (that are part of the second web page) and displays the retrieved web objects as part of the second web page.

Thus, the claims explicitly provide for the use of an object list that identifies web objects. Further, the web objects identified in the list are part of a web page likely to be accessed next. Such use is expressly exemplified in the claims through the requesting of web objects from a server that are identified in the object list.

In rejecting the prior dependent claims 3, 22, and 41 that were directed towards the use of the object list, the Office Action relies on Knight col. 6, lines 18-24 that provides:

In another aspect of this invention, the query handling routine automatically downloads those messages corresponding to information categories previously indicated as of most interest to the particular user. This can occur at the beginning of a session, or during idle periods, so that the user is given a faster response time for messages of interest to him/her.

As can be seen, this text does not describe, teach, suggest, hint, or allude to a list of objects. Instead, the text (and the remainder of Knight) merely refer to categories that are indicated as being of most interest and the downloading of messages that correspond to such categories. Applicants submit that neither the categories or the messages constitute an object list as set forth in the claims. In this regard, the category is not a list of objects that are likely to be accessed next as part of additional web pages. A category is merely an identifier or grouping as described in Knight.

Knight's messages may be classified and located based on predefined information categories (see col. 6, lines 14-17). However, the category itself is not a list of the objects within the category. Instead, it is merely an identifier. Further, such a list is never transmitted, received, or obtained by an applet (as claimed).

Again, the claims provide for first obtaining an object list that identifies web objects likely to be accessed next (as part of a web page likely to be accessed next) and then requesting the web objects from the list. Nowhere in Knight is there a description or suggestion of the use of any such list. In this regard, a query handling routine is used to download messages from certain categories. Firstly, it is unknown if categories of most interest to a particular user are likely to be accessed next. Just because a category is interesting, it may not be likely to be accessed next. Further, even if a message is likely to be accessed next, the claims provide that the web objects are part of a web page that is likely to be requested next. Thus, rather than only claiming the likelihood for accessing an individual message next, the claims provide for web object that are part of an additional web page (and not the same web page) likely to be accessed next. Knight addresses the situation where messages in categories are loaded into the display and not whether an entirely new web page is likely to be accessed next.

Further, the web objects that are retrieved are identified in the list that is obtained at the client by the applet. Accordingly, the list is actually located at the client and the individual objects are downloaded before the client requests the objects. However, Knight's methodology uses a query handling routine that automatically downloads messages for a particular category. There is no indication or suggestion that a list of messages likely to be accessed next is first retrieved followed by the downloading of messages in such a list. Instead, the entire architecture of Knight is directed towards displaying messages in an interface. On its face, it can be clearly seen that the structure, display, and messages of Knight are differentiable from and do not render obvious the detailed limitations of the present claims.

In view of the above, there are clear differences between the present claims and the description in Knight. Such differences establish that the presently amended claims are non-obvious and patentable over Knight.

In addition to the above, the other cited references fail to cure the deficiencies of Knight.

Moreover, the various elements of Applicants' claimed invention together provide operational advantages over Knight, O'Brien, Borman, and Batchelder. In addition, Applicants' invention solves problems not recognized by Knight, O'Brien, Borman, and Batchelder.

Thus, Applicants submit that independent claims 1, 9, 20, 28, 39, and 47 are allowable over Knight, O'Brien, Borman, and Batchelder. Further, dependent claims 2-8, 10-19, 21-27, 29-38, 40-46, and 48-57 are submitted to be allowable over Knight, O'Brien, Borman, and Batchelder in the same manner, because they are dependent on independent claims 1, 9, 20, 28, 39, and 47, respectively, and thus contain all the limitations of the independent claims. In addition, dependent claims 2-8, 10-19, 21-27, 29-38, 40-46, and 48-57 recite additional novel elements not shown by Knight, O'Brien, Borman, and Batchelder.

IV. Conclusion

In view of the above, it is submitted that this application is now in good order for allowance and such allowance is respectfully solicited. Should the Examiner believe minor matters still remain that can be resolved in a telephone interview, the Examiner is urged to call Applicants' undersigned attorney.

Respectfully submitted,

GATES & COOPER LLP
Attorneys for Applicant(s)

Howard Hughes Center
6701 Center Drive West, Suite 1050
Los Angeles, California 90045
(310) 641-8797

Date: October 12, 2005

By: 
Name: Jason S. Feldmar
Reg. No.: 39,187

JSF/mrj